How to fix the Microsoft Office SSO link problem on an Apache webserver

Explanation of the problem

When a user clicks a link in a Microsoft Office document, Office will request the link in the background before opening the browser. This will cause a problem if the link points to a webpage protected by Single Sign-On (SSO).

Normally, when users directly access such a webpage in a browser, they will be redirected to the SSO login page with a “303 See Other” response containing a cookie. This cookie will then identify the newly initiated SSO session and is required for the rest of the sign-on process.

However, when Office requests a URL that gets redirected to another URL (in this case, the SSO login URL), it directly opens the browser with the SSO login URL. As a result, the initial cookie is never set and there is no session for the Identity Provider (IdP) to authenticate. This will result in an error page and the user will likely assume that the link is broken.

Workaround

In order for such a link to work properly in Office, Office must open a browser with the original URL. To achieve this, Office must receive a “200 OK” response and no redirects when requesting the original URL from the Apache webserver.

Apache can identify the client software originating the request by using the User-Agent header. When a request originated from Office is identified by Apache, Apache can be directed to immediately return a “200 OK” response so Office can open the original link in a browser.

Therefore, to fix the problem, do the following on your Apache webserver:

1. Put a simple static html or txt file in a directory on the webserver that has filesystem permissions to be served by Apache. The directory should also be opened to the world with a <Directory> block in Apache’s httpd.conf file. Disabling caching of this html/txt page is recommended. In our example, we are using the txt file at /var/www/html/health.txt, which is used by the load balancers and monitoring. Note that the content of the file is not critical. The following shows the content of a sample txt file.

   ```
   This txt file is used to make Microsoft Office applications work with webpages protected by Single Sign-On (SSO) authentication.
   ```

2. If you are using Apache 2.4, add the following code to httpd.conf. The code requires mod_rewrite and mod_headers. You can make the setting default across all VirtualHosts with “RewriteOptions InheritDown”.

```
# Internally rewrite requests coming from Microsoft Office to a basic 200 page
# This keeps Office from breaking SSO redirects
RewriteEngine On
RewriteCond %{HTTP_USER_AGENT} Microsoft\ Office [OR]
RewriteCond %{HTTP_USER_AGENT} ms-office
RewriteRule ^.*$ /var/www/html/health.txt [L]

#Uncomment this once Apache is upgraded to 2.4 to enable global RewriteRules
#Then place this in httpd.conf to apply to all Virtual Hosts
RewriteOptions InheritDown

# Never cache health check pages
<Directory "/var/www/html">
        Header unset ETag
        Header set Cache-Control "max-age=0, no-cache, no-store, must-revalidate"
        Header set Pragma "no-cache"
        Header set Expires "-1"
</Directory>
```

3. If you are using an earlier version (before 2.4) of Apache, add the following code to httpd.conf for any VirtualHost that has the problem.

```
# Internally rewrite requests coming from Microsoft Office to a basic 200 page
# This keeps Office from breaking SSO redirects
RewriteEngine On
RewriteCond %{HTTP_USER_AGENT} Microsoft\ Office [OR]
RewriteCond %{HTTP_USER_AGENT} ms-office
RewriteRule ^.*$ /var/www/html/health.txt [L]

#Uncomment this once Apache is upgraded to 2.4 to enable global RewriteRules
#Then place this in httpd.conf to apply to all Virtual Hosts
#RewriteOptions InheritDown

# Never cache health check pages
<Directory "/var/www/html">
        Header unset ETag
        Header set Cache-Control "max-age=0, no-cache, no-store, must-revalidate"
        Header set Pragma "no-cache"
        Header set Expires "-1"
</Directory>
```